MASTER’S DEGREE PROGRAM IN

COMPUTER ENGINEERING – COMPUTER SYSTEMS AND NETWORKS

UNIVERSITY OF PISA

The Presentation for the course of

885II 21/22 - MOBILE AND SOCIAL SENSING SYSTEMS

Real-Time Vehicle and Activity Detection using Smartphone Sensors

GROUP 10
Bruno Augusto Casu Pereira De Sousa, Kamran Mehravar
Raul Sevillano Gonzalez, Ayoub El Ourrak

1 June 2022, Pisa

Table of contents

1

Introduction

- ✓ **Application for vehicle and activity detection**
- ✓ **Goals**
- ✓ **Challenges**
- ✓ **Proposed System**

Application for vehicle and Activity detection

Challenges

- ❑ Calculating Activity and vehicle movements in real time with a reliable system.

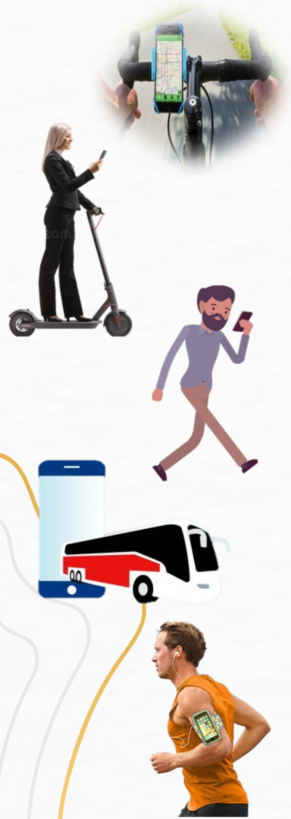
Proposed System

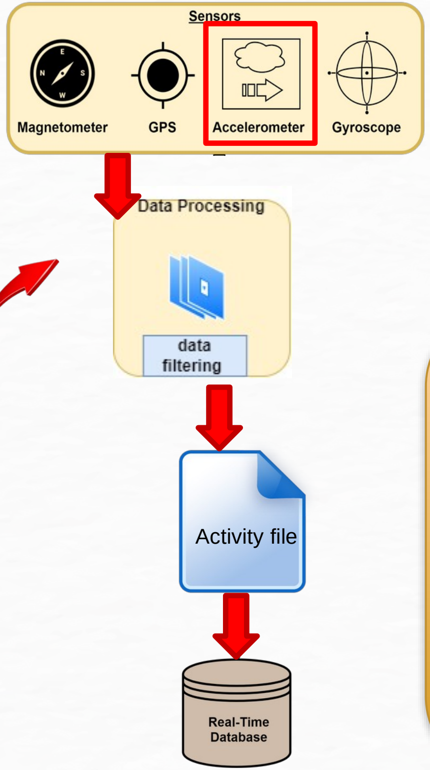
- ❑ A framework to collect data and train a classification model to predict the vehicle and activity mode.

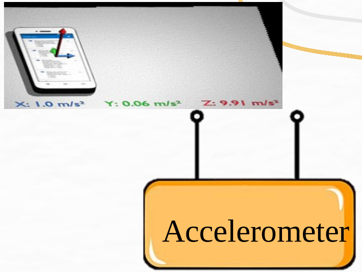
Goals

- ❑ collect sensor data and provide real-time prediction results in a smartphone application.
- ❑ low computational resources and low energy consumption
- ❑ to be able to respond fast to the changes of travelers' vehicle modes.

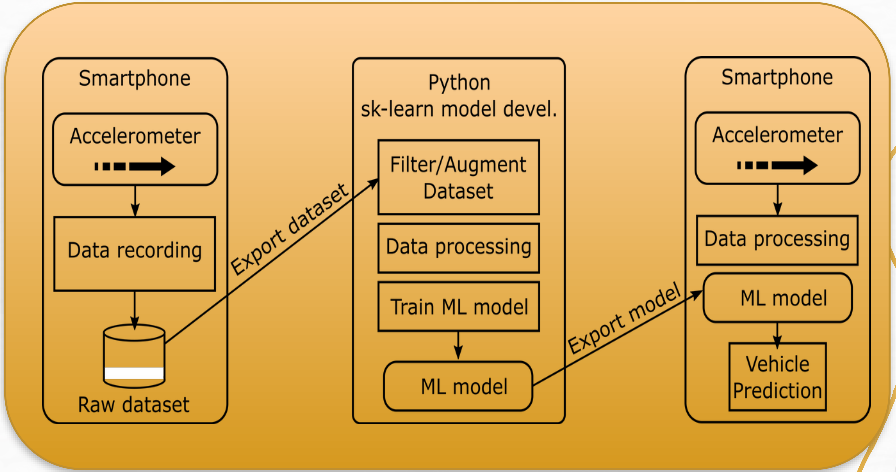
General View of our Model







- An Android application was developed to get the vehicle movement and activity parameters from the Accelerometer sensor in the smartphone



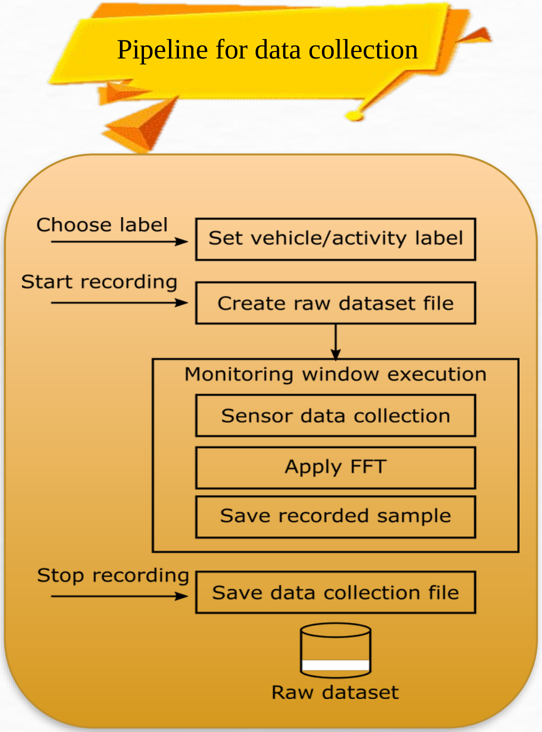
2

Real-Time Vehicle and Activity Detection using Smartphone Sensors

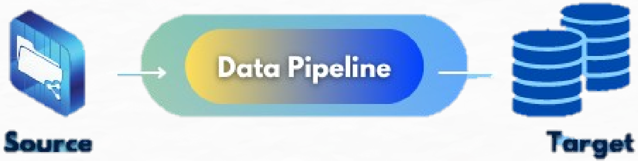
- ✓ **Collecting Data**

Smartphone application for car movement detection: Collecting Data

- The accelerometer data may be acquired using the existing APIs on the Android platform and stored in a temporary dataset to be utilized later for ML training. The data was collected by dividing the samples into collection windows, with the program collecting and storing one frame of data for each time period.



- When performing data collecting, the designed data collection program allows for modification of the applied overlaps. A manual setting is necessary in the code for the window size value, as this may affect how the data is saved. From the Picture depicts the data collection pipeline utilized in the application.

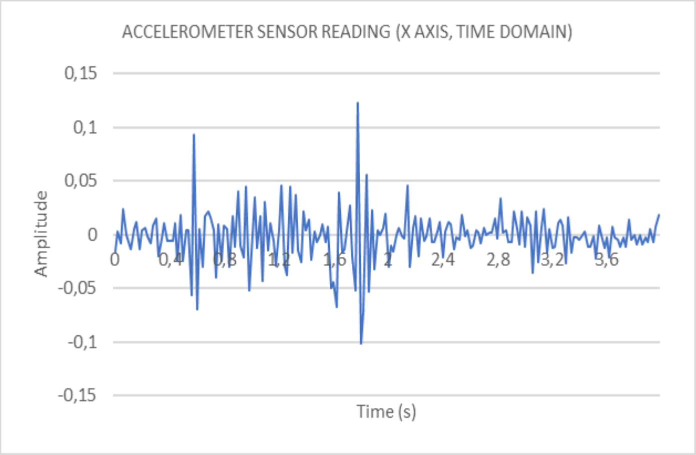


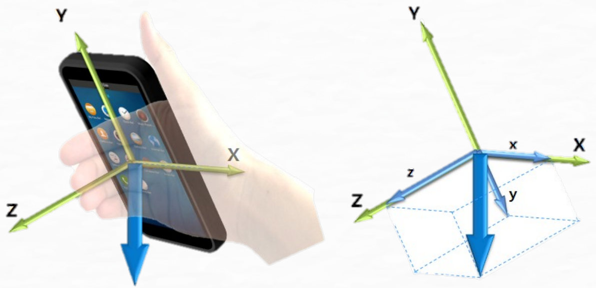
Smartphone application for car movement detection: Collecting Data

- An FFT function is applied to the accelerometer data in the pipeline. This was necessary for the appropriate construction and training of the vehicle and activity detection model given. A MLP type network cannot be trained using data shown in the time domain.

- Picture Below shows an example of data obtained straight from the accelerometer sensor (as supplied by the Android API):

Accelerometer sample data





- The plot shows that the precise traces of the movement that the phone is making may be plotted in different parts of the axis depending on the time frame that the sensor is activated. The model does not converge to a predictive network when trained with this sort of data using an MLP.

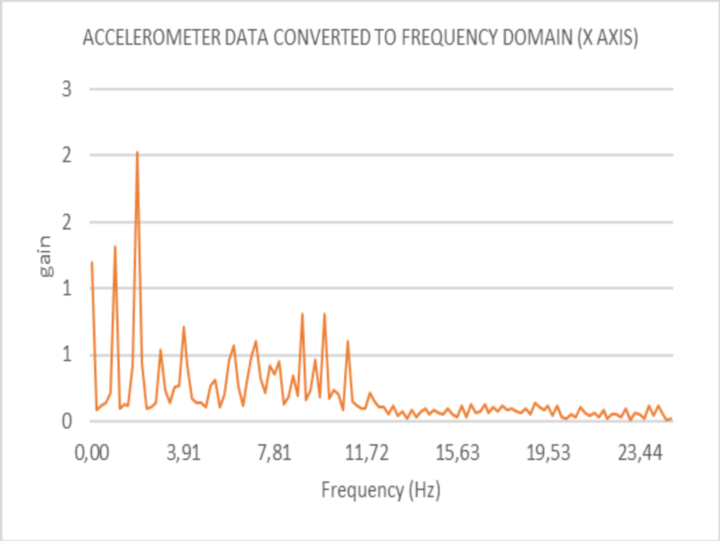
Data collected from the smartphone

Sensor name	Description	Values
Accelerometer	Acceleration profile of a vehicle and activity	accx, accy, accz

Smartphone application for car movement detection: Collecting Data

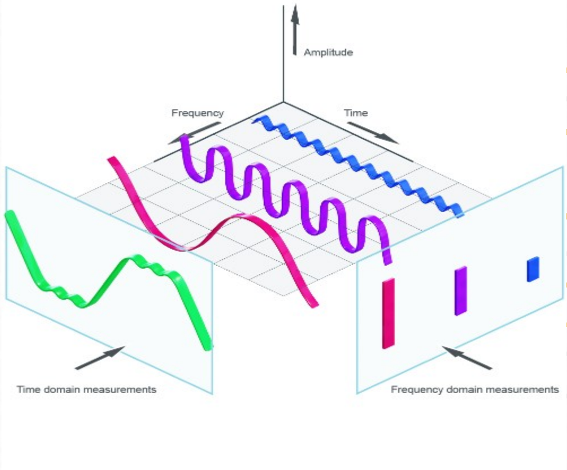
- The data collection monitoring window was adjusted to 256 samples at 20 ms to speed up the FFT computation (5,12 seconds). The FFT function produces a mirrored picture with a frequency range of 0 to 50 Hz.

accelerometer data converted to Frequency Domain



- The data recorded contains the frequency spectrum of the x, y, and z axes. The data were concatenated and transformed to linear values for ease of usage. Another essential aspect of the data gathering is that each orientation is recorded in the same way as the previous one.

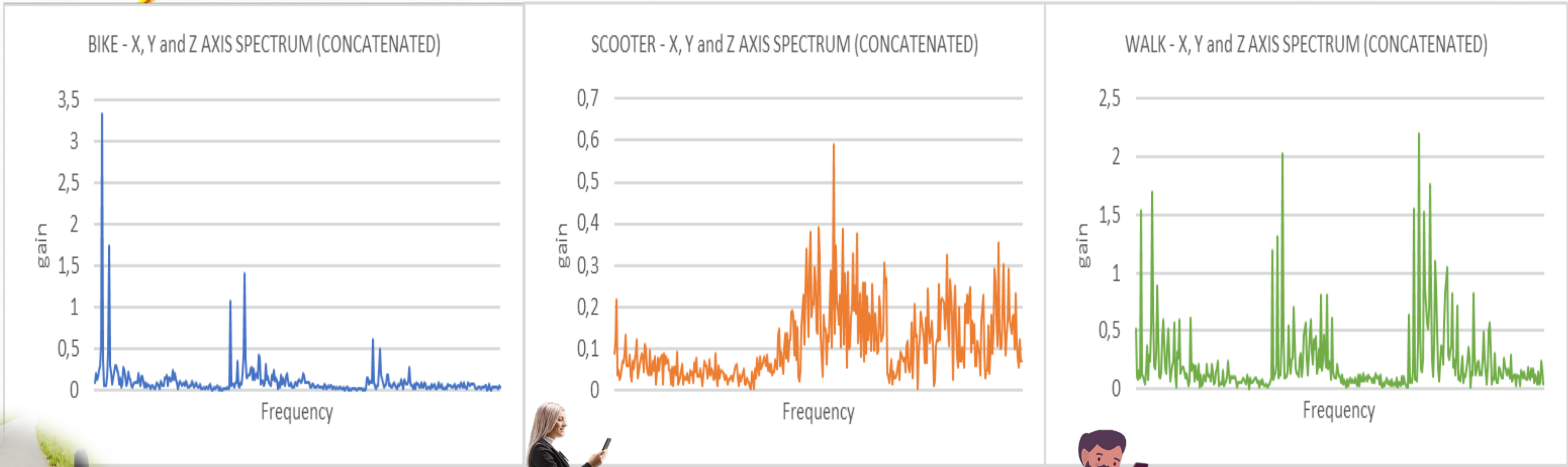
$$gain = 10^{(magnitude/20)}$$



Smartphone application for car movement detection: Collecting Data

❖ An illustration of various frequencies The plots show the signatures detected in a data collection. A varied frequency composition may be seen in the spectrum when looking at the walking plot.

Examples of vehicles frequency spectrum observed in the data collection



3

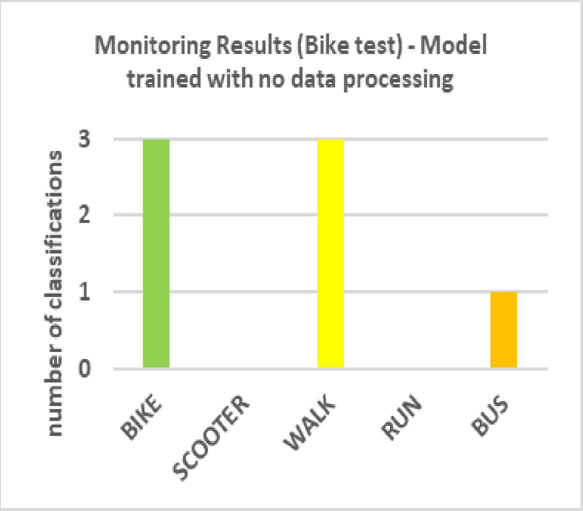
Real-Time Vehicle and Activity Detection using Smartphone Sensors

- ✓ **Preprocessing**

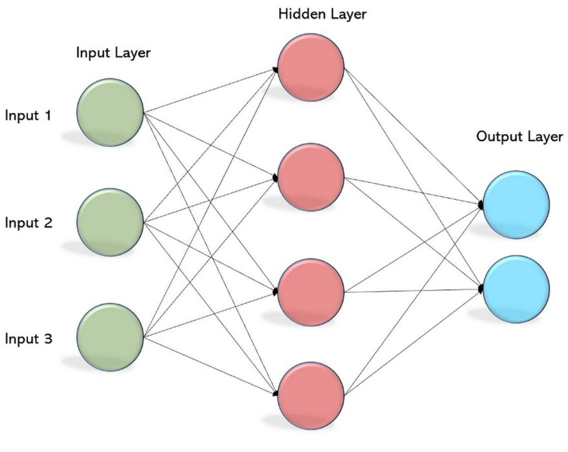
Real-Time Vehicle and Activity Detection using Smartphone Sensors: Preprocessing

❖ An initial training of the MLP classifier model was performed using the provided frequency domain data. The model's total score was high after this initial training. However, during the monitoring tests, it was discovered that the system's effectiveness in recognizing the activity or vehicle was quite low.

Monitoring results using initial MLP



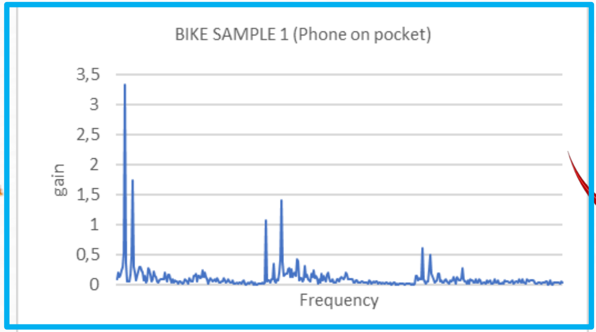
❖ The algorithm misclassified the Bike Test as walking or even as if the user was inside a bus, according to the first data. Further study revealed that the frequency response on the various axes was the source of the misclassification. The frequency peak is shifted to another axis as the phone is positioned in a different way throughout the collection.

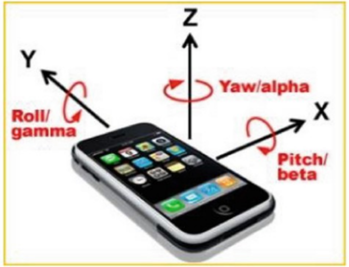


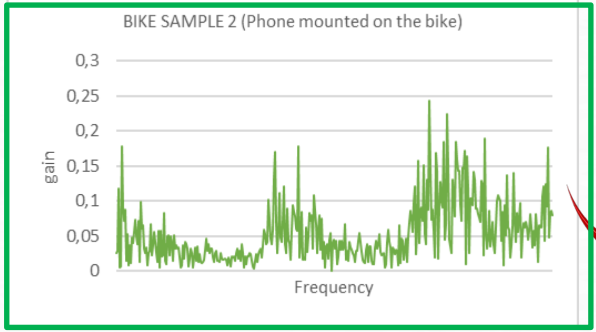
Real-Time Vehicle and Activity Detection using Smartphone Sensors: Preprocessing

- Furthermore, the sample amplitudes might change greatly, making it challenging for an MLP network to train and perform adequately. From the picture depicts two data gathering methods used on a bicycle. The phone was placed in the user's pocket for the first sample, and the result is that the frequency peaks are more prominent on the X axis, with a gain of roughly 3.However, because the phone was now mounted on the Bike, the second sample peaks more exclusively on the Z axis, with a lower gain.

Examples of different frequency spectrums for the same type of vehicle



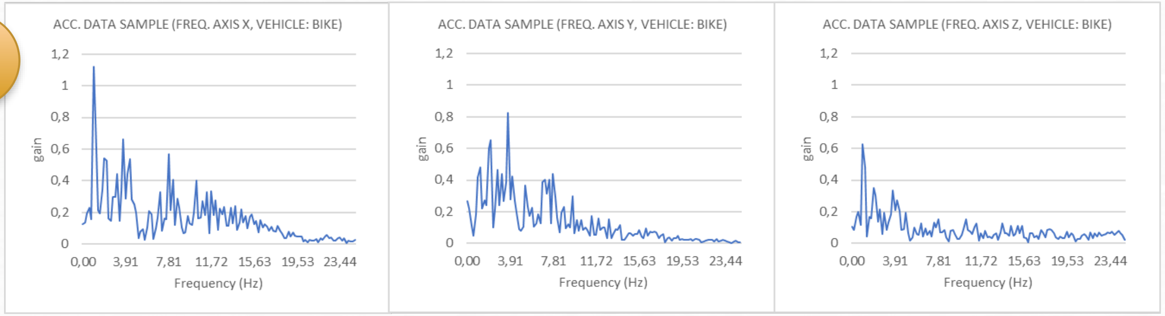




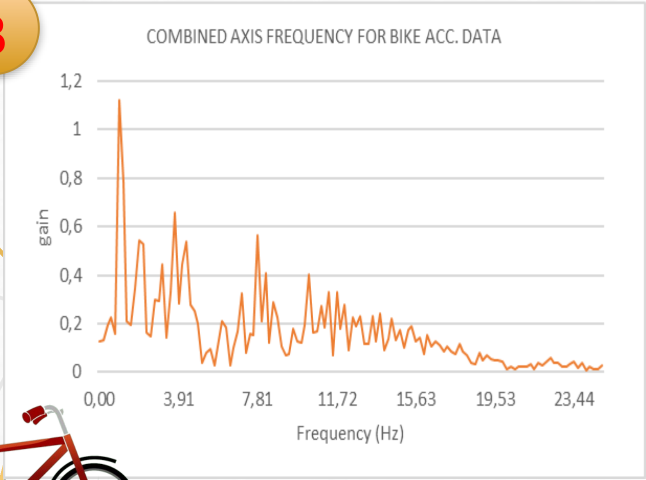
Real-Time Vehicle and Activity Detection using Smartphone Sensors: Preprocessing

❖ Instead of processing each axis individually, the data from all axes is added together in a single array. This means that the predominant frequency (the key frequency to distinguish the vehicle mode) is being highlighted.

A



B



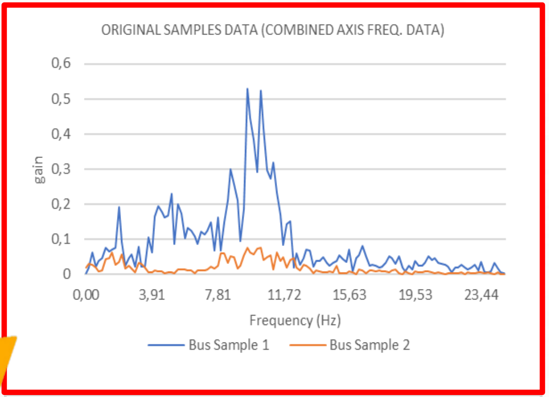
❖ On Figure **A** , a sample of data collected is shown on the three different axes. When combined, the result is the spectrum shown in Figure **B**. As it can be observed the overall shape of the data was maintained and the predominant characteristics were kept

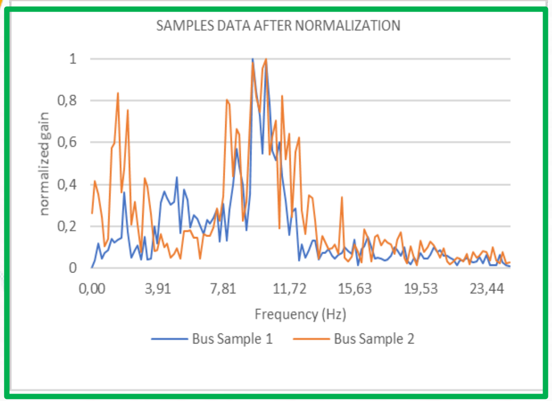
Real-Time Vehicle and Activity Detection using Smartphone Sensors: Preprocessing

- The (FFT) is utilized in the MLP training process to ensure that every data used for training has a consistent maximum value. The minimum value is always zero for each reading acquired from the linear frequency data produced from the FFT conversion, while the maximum value is determined in real-time.

Original and Normalized sample data

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}}$$





- The frequency distribution information is more important than the total gain for each frequency. On the first plot of showing Figure.
- This Figure illustrates an example of the effect of normalization.

- All of the strategies provided in this part for data augmentation are utilized for MLP training and monitoring, therefore the performance improvement in the generated network will be covered in the section ML Model development.

4

Real-Time Vehicle and Activity Detection using Smartphone Sensors

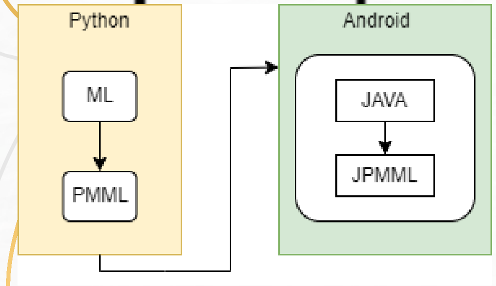
- ✓ **Machine learning**

Real-Time Vehicle and Activity Detection using Smartphone Sensors: Machine learning

scikit learn VS ML Kit

We can achieve a lot of benefits by using scikit-learn library in python instead of ML-kit android for our machine learning part. The main reasons we want to use Scikit-Learn are that it leads to more generality in our system and that our application will not be confined to Android.

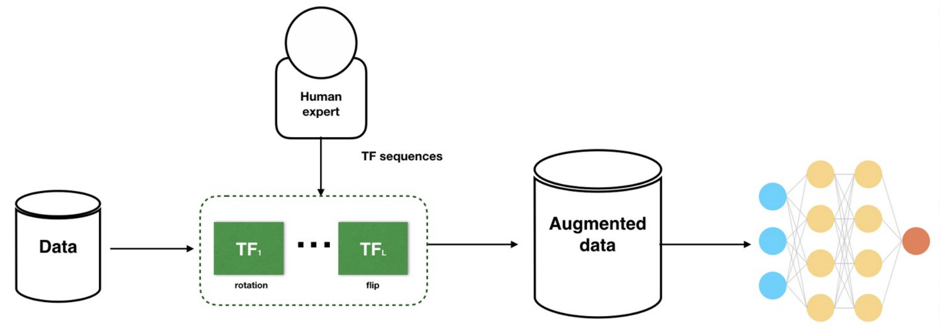
Network model export method



We employed a novel machine learning technique called transfer learning on this project, and we merely transmitted the trained model as a PMML XML file to the mobile to use new data based on prior training for our conclusion. We utilized this technique instead of flask in our application due to our limits on using client and server connections.

Real-Time Vehicle and Activity Detection using Smartphone Sensors: Machine learning

The main dataset used to train the model was gathered from a variety of sources. For each vehicle/activity, a total of 705, 883, 691, 1040, 364 (total: 3683) samples were collected. Using only the collected samples for initial training and testing resulted in substantially overfitted models.



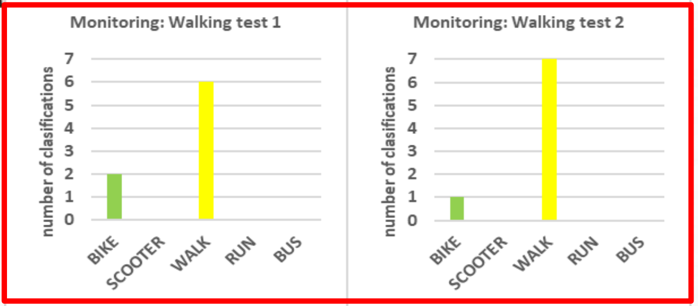
When a machine learning model does not generalize effectively from training data to unknown data, it is said to be overfit. A data augmentation strategy was utilized to expand the overall amount of available data. In general, a large dataset is required for the performance of both ML and Deep Learning models.

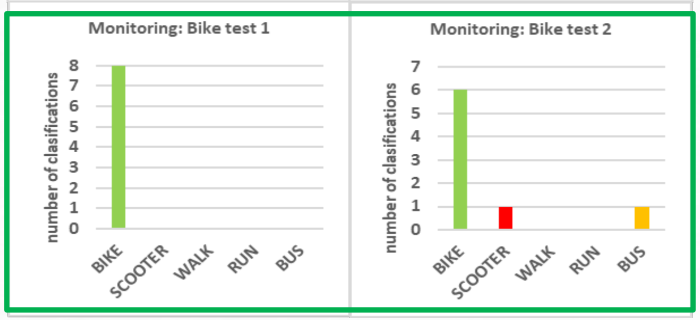
For the time being, the augmentation procedure consists solely of inserting noise into the samples in order to generate fresh data. However, we may improve the model's performance by adding the data we already have. The extended dataset acquired at the end of this data processing stage comprises a total of 10.475 samples.

Real-Time Vehicle and Activity Detection using Smartphone Sensors: Machine learning

Following the enhancement, a filtering algorithm was employed to eliminate any samples with extremely low amplitude that may be deemed noise. The ML dataset used to train the network includes 10.244 samples after the filtering step.

Monitoring Result for walk and bike samples





PREDICTION	BIKE	0,9242	0,0298	0,0275	0,0093	0,0166
	SCOOTER	0,0203	0,9468	0,0085	0,0019	0,0037
	WALK	0,0370	0,0000	0,9535	0,0019	0,0111
	RUN	0,0055	0,0043	0,0000	0,9870	0,0055
	BUS	0,0129	0,0191	0,0106	0,0000	0,9630
		BIKE	SCOOTER	WALK	RUN	BUS
		TRUE LABEL				

The model was trained with a maximum of 100 iterations and only one layer of 100 neurons (default config from sk-learn MLPClassifier network). This training yielded a total accuracy score of 95.5 percent, based on the assessment of over 2.000 samples. The following confusion matrix was created as a result of the research:

5

Real-Time Vehicle and Activity Detection using Smartphone Sensors

- ✓ **Android application**
- ✓ **Data Collection**
- ✓ **Monitoring**

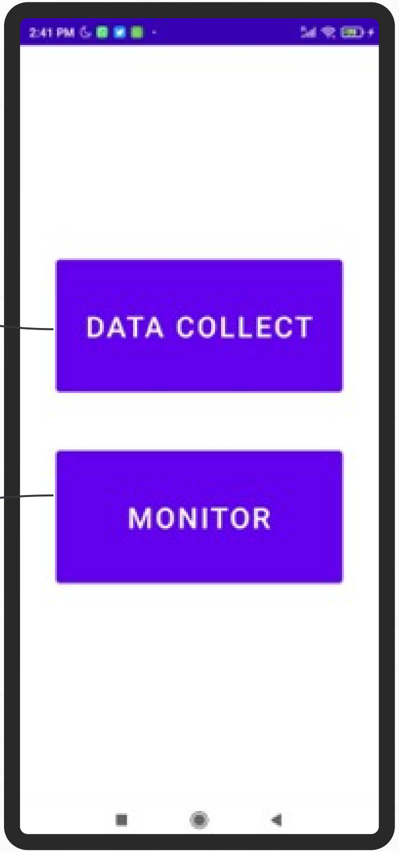
Android Application: Activities

Data Collection

Collect data in real time which are later used to train the machine learning model in the backend

Monitoring

Recognition of the user current activity



Data Collection

Five buttons are used to select in which travel mode we are going to collect the data

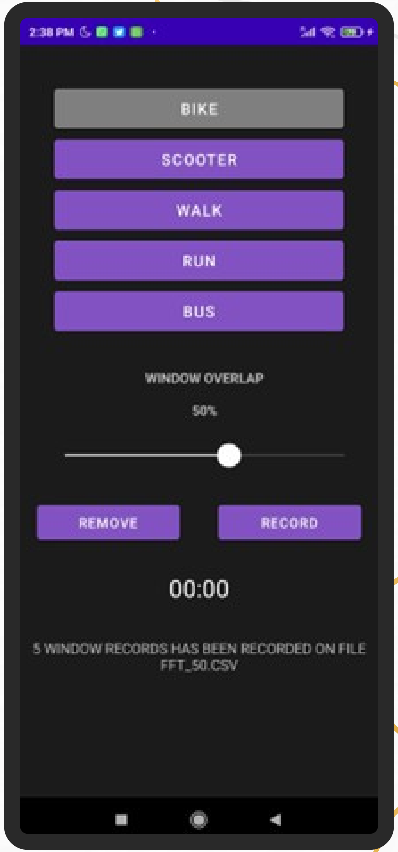
The window overlap ratio can be modified from 0% to 75% using a progress bar

The data are recorded on windows of 256 accelerometers records. After having a fully window the FFT algorithm is applied and magnitude is stored in a csv file

The remove button is used to delete the current data files that have been recorded

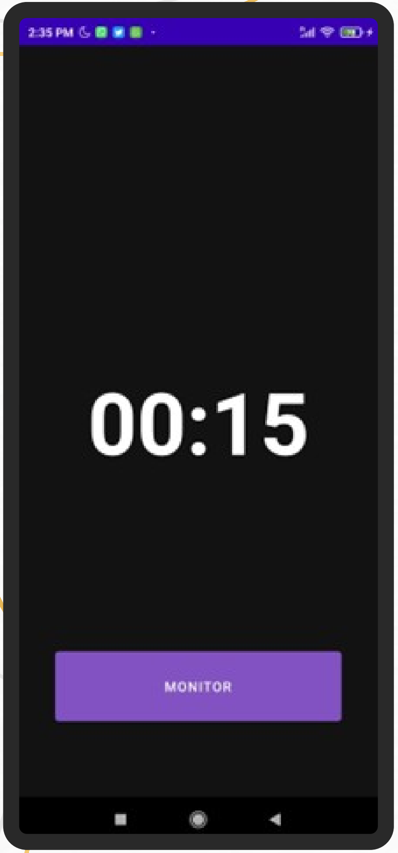
A chronometer shows the amount of time passed since the application has begun collecting data

A text field shows information about application events like remove files or stop recording



Monitoring

Divided in two states

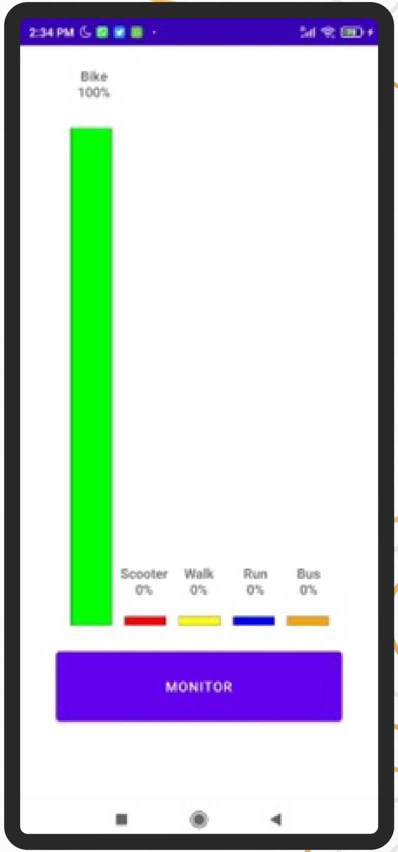


Monitoring:

- After pressing the button the chronometer starts
- At this point the device begins collecting data from accelerometer sensor and process it on same way as on data collection activity
- The application collects data for six windows and store it into a temporal csv file

Showing result:

- The temporal file is taken as input into the machine learning model, which is automatically installed with the application
- This model returns six different results, that predict the travel mode for each data window
- The results are presented in a plot showing the probability that each mode has



6

Real-Time Vehicle and Activity Detection using Smartphone Sensors

✓ **Conclusions**

Conclusions

Using our application the users can detect their current travel mode with an average accuracy of 84.37%.
This accuracy can increase if we collect more data to train our classifier especially with the overlap technique.

That's All Folks!!!